Name: Sahib Bajwa

ID: 107553096

CSCI 3104, Algorithms **Profs. Chen & Grochow**

Final Exam S21–S24 **Spring 2020, CU-Boulder**

Instructions: This quiz is open book and open note. You **may** post clarification questions to Piazza, with the understanding that you may not receive an answer in time and posting does count towards your time limit. Questions posted to Piazza **must be posted as PRIVATE QUESTIONS.** Other use of the internet, including searching for answers or posting to sites like Chegg, is strictly prohibited. Violations of these are grounds to receive a 0 on this quiz. Proofs should be written in **complete sentences. Show and justify all work to receive full credit.**

TIMING: If you are not attempting all the standards in a given quiz, please only use the ordinary amount of time for the number of standards you attempt. For example, if you are only attempting one standard on a 4-standard quiz, please only use 30 min (or 38 for 1.5x, 45 for 2x).

YOU MUST SIGN THE HONOR PLEDGE. Your quiz will otherwise not be graded.
Honor Pledge: On my honor, I have not used any outside resources (other than my notes and book), nor have I given any help to anyone completing this assignment.

Your Name: Sahib Bajwa

Quicklinks: 21 22 23 24

---

21. **Standard 21.** Recall the sequence alignment problem where the cost of *sub* and the cost of *indel* are all 1 (the cost of a noop is 0). Given the following table of optimal cost of aligning the strings EXPON and POLYNO, draw the backward path consisting of backward edges to find the minimal-cost set of edit operations that transforms EXPON to POLYNO. Besides indicating the backward path, you must also give the minimal-cost set of edit operations.

	$-$	P	O	L	Y	N	O
$-$	0	1	2	3	4	5	6
E	1	1	2	3	4	5	6
X	2	2	2	3	4	5	6
P	3	2	3	3	4	5	6
O	4	3	2	3	4	5	5
N	5	4	3	3	4	4	5

(BEGIN YOUR ANSWER ON THE NEXT PAGE.)

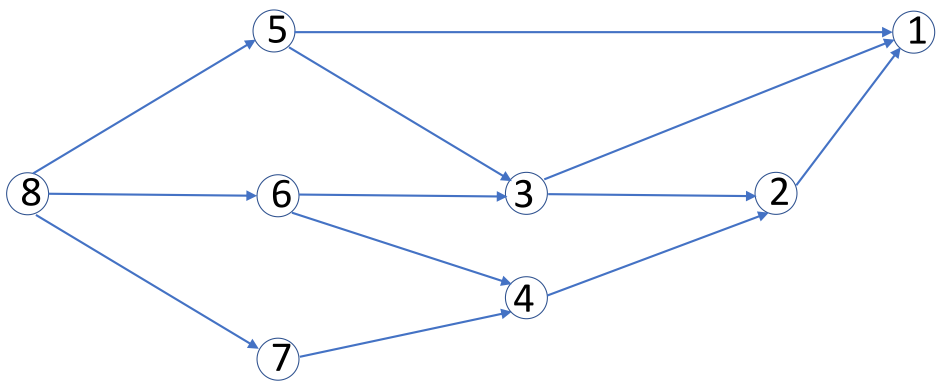
Name: Sahib Bajwa
ID: 107553096

YOUR ANSWER HERE FOR STANDARD 21. (YOU CAN DELETE ALL THIS TEXT IN CAPS.)

IF YOU ARE HANDWRITING AND INSERTING AN IMAGE, SEE THE COMMENTED CODE BELOW IN THE .TEX FILE. PLEASE BE SURE TO ROTATE YOUR IMAGE TO THE CORRECT ORIENTATION (CAN BE DONE IN THE LATEX DIRECTLY; SEE COMMENTS.)

IF YOU'D LIKE TO DRAW YOUR ARROWS IN LATEX, YOU MAY DO SO USING THE COMMENTED OUT TEMPLATE BELOW

22. **Standard 22.** Given the following directed acyclic graph. Use dynamic programming to fill in a table that counts number of paths from each node j to 1, for $1 \le j \le 8$. Note that a single vertex is considered a path of length 0. Hint: you can use a 1-dimensional DP table.



(BEGIN YOUR ANSWER ON THE NEXT PAGE.)

Node	# Paths
1	0
2	1
3	2
4	1
5	3
6	3
7	1
8	7

Hand-checking:
The number of paths from 1 to 1 is 0.
The number of paths from 2 to 1 is 1. This is a direct path.
The number of paths from 3 to 1 is 2. There is one direct path and one path that comes from going to 2.
The number of paths from 4 to 1 is 1. This path comes from going to 2.
The number of paths from 5 to 1 is 3. There is one direct path and two paths that comes from going to 3.
The number of paths from 6 to 1 is 3. There are two paths that come from going to 3 and one path that comes from going to 4.
The number of paths from 7 to 1 is 1. This path comes from going to 4.
The number of paths from 8 to 1 is 7. There are three paths that come from going to 5, three paths that come from going to 6, and one path that comes from going to 7.

Name: Sahib Bajwa
ID: 107553096

23. **Standard 23.** The Spanning Tree problem is defined as follows.

- Instance: $(G, w : E(G) \to \mathbb{R}_{>0}, k)$ where G is a simple, connected graph, each edge e has weight $w(e) > 0$, and a number $k > 0$.
- Decide: Does G have a spanning tree weight at most k?

Show that Spanning Tree $\in$ P. You are welcome and encouraged to use algorithms we have previously studied, as well as any well-known facts about the complexity of said algorithms.

(BEGIN YOUR ANSWER ON THE NEXT PAGE.)

YOUR ANSWER HERE FOR STANDARD 23. (YOU CAN DELETE ALL THIS TEXT IN CAPS.)

IF YOU ARE HANDWRITING AND INSERTING AN IMAGE, SEE THE COMMENTED CODE BELOW IN THE .TEX FILE. PLEASE BE SURE TO ROTATE YOUR IMAGE TO THE CORRECT ORIENTATION (CAN BE DONE IN THE LATEX DIRECTLY; SEE COMMENTS.)

24. **Standard 24.** Consider the Integer Factorization problem, which is in NP.

- Instance: $n, k > 0$ integers.
- Decision: Does n have a factor d, where $2 \leq d \leq k$?

The RSA cryptosystem, which is used to secure our online transactions, relies on the difficulty of the Integer Factorization problem (which is not known to be either in P nor NP-Complete). In other words, if we can factor integers efficiently, we can break RSA. Suppose we have a polynomial time algorithm A to solve an NP-Complete problem L. Explain carefully why this would break RSA.

(BEGIN YOUR ANSWER ON THE NEXT PAGE.)

YOUR ANSWER HERE FOR STANDARD 24. (YOU CAN DELETE ALL THIS TEXT IN CAPS.)

IF YOU ARE HANDWRITING AND INSERTING AN IMAGE, SEE THE COMMENTED CODE BELOW IN THE .TEX FILE. PLEASE BE SURE TO ROTATE YOUR IMAGE TO THE CORRECT ORIENTATION (CAN BE DONE IN THE LATEX DIRECTLY; SEE COMMENTS.)